

NAME

CURLINFO_RETRY_AFTER – returns the Retry–After retry delay

SYNOPSIS

```
#include <curl/curl.h>
```

```
CURLcode curl_easy_getinfo(CURL *handle, CURLINFO_RETRY_AFTER,
                           curl_off_t *retry);
```

DESCRIPTION

Pass a pointer to a `curl_off_t` variable to receive the number of seconds the HTTP server suggests the client should wait until the next request is issued. The information from the "Retry–After:" header.

While the HTTP header might contain a fixed date string, the *CURLINFO_RETRY_AFTER(3)* always returns the number of seconds to wait – or zero if there was no header or the header could not be parsed.

This option used to return a negative wait time if the server provided a date in the past. Since 8.12.0, a negative wait time is returned as zero. In any case we recommend checking that the wait time is within an acceptable range for your circumstance.

DEFAULT

Zero if there was no header.

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
int main(void)
{
    CURL *curl = curl_easy_init();
    if(curl) {
        CURLcode result;
        curl_easy_setopt(curl, CURLOPT_URL, "https://example.com");
        result = curl_easy_perform(curl);
        if(result == CURLE_OK) {
            curl_off_t wait = 0;
            curl_easy_getinfo(curl, CURLINFO_RETRY_AFTER, &wait);
            if(wait)
                printf("Wait for %" CURL_FORMAT_CURL_OFF_T " seconds\n", wait);
        }
        curl_easy_cleanup(curl);
    }
}
```

AVAILABILITY

Added in curl 7.66.0

RETURN VALUE

curl_easy_getinfo(3) returns a `CURLcode` indicating success or error.

`CURLE_OK` (0) means everything was OK, non-zero means an error occurred, see *libcurl-errors(3)*.

SEE ALSO

CURLOPT_HEADERFUNCTION(3), CURLOPT_STDERR(3), curl_easy_header(3)